



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/871,345	05/30/2001	Radu V. Ionescu	MS163104.2/40062.116USUI	9575
45979	7590	01/11/2006	EXAMINER	
PERKINS COIE LLP/MSFT P. O. BOX 1247 SEATTLE, WA 98111-1247			RUTTEN, JAMES D	
			ART UNIT	PAPER NUMBER
			2192	
DATE MAILED: 01/11/2006				

Please find below and/or attached an Office communication concerning this application or proceeding.

<b>Office Action Summary</b>	<b>Application No.</b> 09/871,345	<b>Applicant(s)</b> IONESCU, RADU V.	
	<b>Examiner</b> J. Derek Rutten	<b>Art Unit</b> 2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 08 July 2005.
- 2a) ☒ This action is **FINAL**.                      2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-27 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-27 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All    b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)  | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)                                   | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)             |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)<br>Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____  |

### **DETAILED ACTION**

1. This action is responsive to Applicant's amendment dated 9/14/2004, responding to the 6/16/2004 Office action provided in the rejection of claims 1-15, wherein claims 2, 8, and 12 have been amended, and new claims 16-27 have been added. Claims 1-27 remain pending in the application and have been fully considered by the examiner.

2. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the date of this final action.

### ***Response to Arguments***

3. Applicant's amendments have overcome the objections to the drawings, specification, and claims, and also the rejection under 35 U.S.C. 112. Likewise, these objections and rejections are withdrawn.

Art Unit: 2192

4. Applicant's arguments filed 9/14/2004 have been fully considered but they are not persuasive. Applicant essentially argues (esp. bottom of page 10 – page 11) that Kobayashi does not retrieve from an input dialog a selected method, or selecting parameters for the selected method. However, further review of Kobayashi shows that a bean compiler produces proxy components for all components as well as component methods. These proxy components are then manipulated in a visual builder. See column 8 lines 10-18:

As previously mentioned, the bean compiler 208 converts each component into proxy components 210, where **each method in the component is converted into a separate proxy component of the components 210. The proxy components 210 can be manipulated with the help of a conventional visual builder 214** which has been provided with a bean visual environment add-on that allows the proxy components to be recognized by the visual builder. (emphasis added)

Kobayashi further discloses manipulation of the beans and their attributes in terms of methods and their parameters in a visual builder. See column 9 lines 30-39:

In accordance with the principles of the invention, visual builder 214 includes a bean visual environment add-on 212, which allows the proxy beans 210 to appear directly on the builder's object palette. The proxy beans can therefore be manipulated in a conventional fashion using builder 214. Since the proxy beans include the parameters of the methods as attributes, **the method parameters can be manipulated directly** by changing the attributes of the proxy beans in order to generate a bean-based application 216. (emphasis added)

Further, Figs. 11, 14, and 17 show examples of input dialogs that appear in a visual builder that allows a user to retrieve a method and obtain parameters. Applicant essentially argues that Kobayashi selects a composite component as opposed to a method of an object (see top of page 11 of the response). However, as pointed out in the cited passages from Kobayashi above, the proxy components relate directly to object methods. Selection of a proxy component is equivalent to selection of an object method. It is the method that will be exercised and tested, not the proxy component. Thus, the limitations are disclosed by Kobayashi, and Applicant's arguments are not persuasive.

Art Unit: 2192

5. Applicant essentially argues (see paragraph 3 page 11) that Kobayashi does not teach “repeated invocation”. However, Kobayashi discloses manipulation and testing of a plurality of components (see column 18 lines 53-58) which would require a “repeated invocation”, otherwise only one component could be invoked. Thus, this argument is not convincing.

***Claim Rejections - 35 USC § 101***

6. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

7. Claims 16-27 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Newly added independent claims 16 and 22 are directed to a “computer-readable medium”. However, computer readable media, as defined in the originally filed specification on page 6 lines 13-20 includes such non-tangible media as a “carrier wave”. Statutory subject matter includes tangible media such as computer readable storage media, as described in the specification on page 6 lines 9-13. Claims 17-21 and 23-27 depend upon claims 16 and 22, respectively, and are rejected for the same reasons set forth above.

***Claim Rejections - 35 USC § 112***

8. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Art Unit: 2192

9. Claims 16-27 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

10. Claims 16-27 are rejected under 35 U.S.C. 112, second paragraph, as being incomplete for omitting essential elements, such omission amounting to a gap between the elements. See MPEP § 2172.01. The omitted elements are: support of the preamble, which is directed to a “computer-readable medium”, by the body of the claim. The body of the claim merely recites method steps, but does not describe any elements that could be found on a computer-readable medium. Such elements might include instructions encoded on the computer-readable medium that would provide the functionality of the method.

***Claim Rejections - 35 USC § 102***

11. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

12. Claims 1-10, and 16-18 are rejected under 35 U.S.C. 102(e) as being anticipated by prior art of record U.S. Patent 6,633,888 to Kobayashi (hereinafter referred to as “Kobayashi”).

As per claim 1, Kobayashi discloses:

Art Unit: 2192

*A method for reviewing operation of software objects in a computer program*

(column 24 lines 20-62), *the software objects, methods on the objects and parameters for the methods are stored in a library, the method comprising the acts of:*

*retrieving an object for review from an object store to provide a selected object*

See column 22 lines 38-40:

In step 2002, the **bean** to be tested is **loaded into the tester** described above using the visual environment add-on, also as discussed above

also column 11 lines 18-20:

VEA 700 allows access by the visual builder 708 to beans contained in **JAR file 702**, which beans may be the beans created by the bean creator as discussed above

*retrieving from an input dialog a selected method for exercising the selected*

*object* See column 22 lines 41-42:

Next, in step 2004, the bean is displayed in the workspace window by **selecting it from the palette**.

Also column 22 lines 30-32:

Testing a **composite bean** typically involves testing the operation of its **methods** and bound properties, which, as discussed above include the parameters of the methods.

*obtaining selected parameters for use in the method for exercising the selected*

*object* See column 22 lines 49-53:

Further, as discussed previously, **the method parameters** of the original bean are exposed by the proxy components created from the methods of that bean. Consequently, each method can be tested fully.

also column 8 lines 23-27:

Each composite component in the application 216 can be tested within the visual builder by means of the universal transport API 206 which allows the code which implements the underlying **objects** and components 202 to be **exercised** under control of the proxy components.

*exercising the selected object with the selected method using the selected parameters so that the operation of the software object is reviewed* See column 22 lines 46-49:

As previously mentioned, when the methods of proxy beans are invoked, they use the universal transport mechanism to invoke the actual component code in order to test the method as set forth in step 2008.

As per claim 2, the above rejection of claim 1 is incorporated. Kobayashi further discloses: *wherein the library is a type library* (column 8 lines 60-63) *and the method further comprises: creating, using information stored in the type library, an instance of the object for review; and storing the instance of the object in the object store* (column 22 lines 12-15).

As per claim 3, the above rejection of claim 1 is incorporated. Kobayashi further discloses: *wherein the act of exercising creates another object that may be exercised and the method further comprises: storing the another object in the object store* (column 12 line 56 – column 13 line 13). All further limitations have been addressed in the above rejection of claim 2.

As per claim 4, the above rejection of claim 1 is incorporated. Kobayashi further discloses: *wherein the library is a type library* (column 8 lines 60-63) *and the act of obtaining comprises: displaying an input dialog for a user to chose parameters using information stored in the type library or already retrieved for use with the selected method exercising the selected object* (FIG. 11 element 1160).



As per claim 5, the above rejection of claim 4 is incorporated. All further limitations have been addressed in the above rejection of claim 1.

As per claim 6, Kobayashi discloses:

*A method for testing software objects (column 24 lines 20-62) comprising:  
receiving input dialog selections that specify a selected object and a selected  
method for operation on the object during testing (FIG. 11; also column 22 lines 38-40 as  
cited in the above rejection of claim 1);*

*detecting whether testing of the selected object is complete See column 23 lines 1-  
3:*

If the bean did not pass the test, the operator may then choose to edit the bean as set forth in step 2014.

*if testing of the selected object is not complete, calling the selected method on the  
selected object and displaying an input dialog for parameters to provide selected  
parameters for use by the selected method See column 18 lines 12-15:*

The properties window 1155 displays to the user **properties** 1160 and the corresponding property value 1165 associated with a **bean selected** in the workspace window 1140.

*exercising the selected method on the selected object using the selected  
parameters from the input dialog so that software objects may be tested for operation  
with the selected method using the selected parameters See column 23 lines 3-5:*

The method then proceeds back to step 2006 where the bean is rerun to retest it.

As per claim 7, the above rejection of claim 6 is incorporated. Kobayashi further discloses: *creating a library store storing object information defining elements of each software object* (column 8 lines 60-63); *and retrieving object information from the library store including methods on each object* (column 18 lines 11-20).

As per claim 8, the above rejection of claim 6 is incorporated. Kobayashi further discloses: *wherein the input dialog selections include a create selection indicating an instance of the selected object is to be created and the method further comprises: creating an instance of the selected object* (column 22 lines 12-15 and 43-49).

As per claim 9, the above rejection of claim 8 is incorporated. Kobayashi further discloses: *wherein the act of exercising comprises: retrieving selected parameters chosen from the input dialog for parameters; and invoking on the instance of the selected object the selected method and using the selected parameters chosen from the input dialog to exercise the instance of the selected object* (column 22 lines 46-53).

As per claim 10, the above rejection of claim 9 is incorporated. Kobayashi further discloses: *interpreting the result of exercising the instance of the selected object* (column 22 line 66 – column 23 line 3).

In regard to claim 16, Kobayashi discloses:

Art Unit: 2192

*A computer-readable medium containing instructions for controlling a computer system to test a software object (See column 23 lines 6-34), by a method comprising:*

*instantiating an object; See column 9 lines 40-44:*

The bean-based application 216 can be tested within the visual environment by using the universal transport API 206 to access the Java objects instantiated from a selected class called the "target" class in the class and bean implementations 202.

*and*

*exercising the instantiated object by repeatedly: See column 18 lines 53-58:*

In step 1210, an operator selects components from the palette 1125 which are to comprise the ultimate composite component. In the illustrative embodiment, the **components can be selected from the palette 1125 by holding down the "shift" key and clicking on each component icon in turn from the palette 1125, as desired.**

Note that the language in the above passage describes a process of display and selection for plural components. This implies a repeated display and selection of components. Further, column 23 lines 3-5 describes the repeated selection and testing of components.

*displaying to a user a list of methods of the object; See column 9 lines 30-39:*

In accordance with the principles of the invention, visual builder 214 includes a bean visual environment add-on 212, which **allows the proxy beans 210 to appear directly on the builder's object palette.** The proxy beans can therefore be manipulated in a conventional fashion using builder 214. Since the proxy beans include the parameters of the methods as attributes, the method parameters can be manipulated directly by changing the attributes of the proxy beans in order to generate a bean-based application 216. (emphasis added)

*receiving from the user a selection of a method; See column 9 lines 30-39:*

In accordance with the principles of the invention, visual builder 214 includes a bean visual environment add-on 212, which allows the proxy beans 210 to appear directly on the builder's object palette. **The proxy beans can therefore be manipulated in a conventional fashion using builder 214.** Since the proxy beans include the parameters of the methods as attributes, the method

Art Unit: 2192

parameters can be manipulated directly by changing the attributes of the proxy beans in order to generate a bean-based application 216. (emphasis added)

*receiving from the user a selection of parameters for the method;* See

column 9 lines 30-39:

In accordance with the principles of the invention, visual builder 214 includes a bean visual environment add-on 212, which allows the proxy beans 210 to appear directly on the builder's object palette. The proxy beans can therefore be manipulated in a conventional fashion using builder 214. Since the proxy beans include the parameters of the methods as attributes, **the method parameters can be manipulated directly** by changing the attributes of the proxy beans in order to generate a bean-based application 216. (emphasis added)

*and*

*invoking the selected method of the instantiated object passing the selected*

*parameters* See column 9 lines 44-49:

In particular, universal transport API 206 operates under control of constructor and method objects instantiated by the proxy beans 210 within bean-based application 216 to **call the appropriate constructors and methods** for the target class in the implementation code 202 as will be hereinafter explained in detail. (emphasis added)

*until the methods of the instantiated object are tested.* See column 9 lines 40-44:

The bean-based application 216 can be tested within the visual environment by using the universal transport API 206 to access the Java objects instantiated from a selected class called the "target" class in the class and bean implementations 202.

In regard to claim 17, the above rejection of claim 16 is incorporated. All further limitations have been addressed in the above rejection of claim 4.

In regard to claim 18, the above rejection of claim 16 is incorporated. All further limitations have been addressed in the above rejection of claim 1

***Claim Rejections - 35 USC § 103***

13. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

14. Claim 11, and 20-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kobayashi as applied to claims 10 and 16 above, and further in view of prior art of record U.S. Patent 6,067,639 to Rodrigues et al. (hereinafter referred to as "Rodrigues").

As per claim 11, the above rejection of claim 10 is incorporated. Kobayashi does not expressly disclose *logging the result interpreted by the act of interpreting*.

However, in an analogous environment, Rodrigues teaches that a log can be used to store test results (column 9 lines 28-31).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use a log to store results of a test. One of ordinary skill would have been motivated to keep a detailed record of test results to compare and track the progress of a project.

In regard to claim 20, the above rejection of claim 16 is incorporated. All further limitations have been addressed in the above rejection of claim 11.

Art Unit: 2192

In regard to claim 21, the above rejection of claim 20 is incorporated. All further limitations have been addressed in the above rejection of claim 11.

In regard to claim 22, Kobayashi discloses:

*A computer-readable medium containing instructions for controlling a computer system to test software objects, (See column 23 lines 6-34) by a method comprising:*  
*providing entries that specify an object, a method of the object, and a parameter of*  
*the method; and for each entry, instantiating the object of the entry; invoking the method*  
*of the entry of the instantiated object with the parameter of the entry; See column 9 lines*  
30-39:

In accordance with the principles of the invention, visual builder 214 includes a bean visual environment add-on 212, which allows the proxy beans 210 to appear directly on the builder's object palette. The proxy beans can therefore be manipulated in a conventional fashion using builder 214. Since the proxy beans include the parameters of the methods as attributes, the method parameters can be manipulated directly by changing the attributes of the proxy beans in order to generate a bean-based application 216.

Also see column 9 lines 44-49:

In particular, universal transport API 206 operates under control of constructor and method objects instantiated by the proxy beans 210 within bean-based application 216 to call the appropriate constructors and methods for the target class in the implementation code 202 as will be hereinafter explained in detail.

All further limitations have been addressed in the above rejection of claim 11.

In regard to claim 23, the above rejection of claim 22 is incorporated. Kobayashi does not expressly disclose: *wherein the entries are provided in a file*. However, Rodrigues teaches that entries can be provided in a file. See column 9 lines 27-31.

In regard to claim 24, the above rejection of claim 22 is incorporated. All further limitations have been addressed in the above rejection of claim 1.

In regard to claim 25, the above rejection of claim 22 is incorporated. All further limitations have been addressed in the above rejection of claim 4.

In regard to claim 26, the above rejection of claim 22 is incorporated. Kobayashi further discloses: *wherein an entry includes multiple parameters*. See column 9 lines 5-19.

15. Claims 12-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kobayashi and Rodrigues in view of prior art of record "Comparing Microsoft Transaction Server to Enterprise JavaBeans" by Microsoft (hereinafter referred to as "MTS").

As per claim 12, Kobayashi discloses:

*parsing object information into methods and parameters* See column 8 lines 35-37:

Bean compiler 302 **parses** the code and extracts the relevant **methods and parameters**.  
*storing the methods and parameters in a library store* See column 8 lines 59-63:

As the beans are created they are inserted into a JAR file 324 by JAR File loader 322. Once all of the beans have been created, another module, the manifest file creator 308, of bean compiler 306 produces a complete **manifest file**, with an ".mf" suffix.

*detecting an input selection indicating each object to be exercised and the method to exercise the object* See column 22 lines 30-22:

Testing a **composite bean** typically involves testing the operation of its **methods** and bound properties, which, as discussed above include the parameters of the methods.

also column 22 lines 41-42:

Next, in step 2004, the bean is displayed in the workspace window by **selecting it from the palette**.

*creating an instance of the object is to be exercised* See column 22 lines 12-15:

With that information specified, in step 1908, the bean tester creates the source code for the composite component, and creates manifest and JAR files for the component in step 1910.

*getting the method and parameters chosen for use with the method to exercise the instance of the object* See column 22 lines 49-53:

Further, as discussed previously, the **method parameters** of the original bean are exposed by the proxy components created from the methods of that bean. Consequently, each method can be tested fully.

also column 8 lines 23-27:

Each composite component in the application 216 can be tested within the visual builder by means of the universal transport API 206 which allows the code which implements the underlying **objects** and components 202 to be **exercised** under control of the proxy components.

*invoking the method with a chosen parameters to exercise the instance of the object to be exercised* See column 22 lines 46-49:

As previously mentioned, when the methods of proxy beans are invoked, they use the universal transport mechanism to invoke the actual component code in order to test the method as set forth in step 2008.

*repeating the detecting, creating, getting, <and> invoking ...for use in debugging and adjusting the operation of the objects* See column 23 lines 3-5:

The method then proceeds back to step 2006 where the bean is rerun to retest it.



Art Unit: 2192

Kobayashi does not expressly disclose creating a log record of objects that are exercised, or the use of COM objects.

However, Rodrigues teaches a method of saving test results to a log file See column 9 lines 28-31:

The test operations object library 406 receives pre-recorded pseudo-random test suite files or playback files 400, and logs all testing procedures to playback files and results to log files 402.

Also, in an analogous environment, MTS teaches that beans (used by Kobayashi) are analogous to MTS COM objects See page 2 last paragraph:

Each **bean** exposes its own Home interface, **analogous** to the **COM** IclassFactory interface, allowing a client to create instances of specific classes.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Rodrigues' teaching of log files with MTS' teaching of COM objects in the test system of Kobayashi. One of ordinary skill would have been motivated to test component-based applications to ensure proper execution. Also, one of ordinary skill would have been motivated to store a log of the testing to permit later examination or reproduction of a failed sequence.

As per claims 13-15, the above rejection of claim 12 is incorporated. All further limitations have been addressed in the above rejections of claims 2, 6, and 10, respectively.

16. Claim 19 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kobayashi.

In regard to claim 19, the above rejection of claim 16 is incorporated. Kobayashi does not expressly disclose: *repeating the instantiating and exercising for another object*. However, Kobayashi is based in an environment made up of numerous objects. See Fig. 2 element 204. It would have been obvious to one of ordinary skill in the art at the time the invention was made to repeat the instantiating and exercising for another object. One of ordinary skill would have been motivated to test more than one object so that the system could be used for testing a plurality of classes (see column 4 lines 61-63).

17. Claim 27 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kobayashi and Rodrigues as applied to claim 22 above, and further in view of U.S. Patent 6,519,605 to Gilgen et al. (hereinafter "Gilgen").

In regard to claim 27, the above rejection of claim 22 is incorporated. Kobayashi does not expressly disclose: *not instantiating an object that has already been instantiated*. However, in an analogous environment, Gilgen teaches that an object that has already been instantiated does not need to be instantiated. See column 15 lines 10-13. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Gilgen's teaching of object instantiation with Kobayashi's objects. One of ordinary skill would have been motivated to reuse an existing object instance in order to avoid the computational expense of creating a new one.

Art Unit: 2192

***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to J. Derek Rutten whose telephone number is (571) 272-3703. The examiner can normally be reached on T-F 6:00 - 4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

jdr



TUAN DAM  
SUPERVISORY PATENT EXAMINER